| Notice of Allowability | Application No. | Applicant(s) | |
|---|---|---|---|
| | 09/751,105 | EYLON ET AL. | |
| | Examiner | Art Unit | |
| | Haresh Patel | 2154 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--*

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *8/21/06*.

2. ☒ The allowed claim(s) is/are *1-6,9-25,28-40 and 43-52*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
**THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.**

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS ( as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review ( PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

**Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).**

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

**Attachment(s)**

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 3/2/06, 2/2/01
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date *09/13/2006* .
7. ☒ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

## EXAMINER'S AMENDMENT

1. Claims 1-6, 9-25, 28-40, 43-52 are presented for examination. Claims 7, 8, 26, 27, 41, 42, 53-68 are cancelled.

2. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

3. Authorization for this examiner's amendment was given in a telephone interview with Mr. Jordan M. Becker on September 13, 2006.

### *Amendments to the Specification*

4. Line 8 of page 1, before "entitled" please insert --, now patent number 6,311,221,--

5. Line 12 of page 1, before ";" please insert --, now abandoned--

6. Line 19 of page 4, before "entitled" please insert --, now patent number 6,311,221,--

### *Amendments to the Claims*

7. Please amend claims 1, 9, 19, 28, 38, 43; and cancel claims 7, 8, 26, 27, 41, 42, 53-68, as attached.

### *Allowable Subject Matter*

8. Claims 1-6, 9-25, 28-40, 43-52 are allowed.

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Haresh Patel whose telephone number is (571) 272-3973. The examiner can normally be reached on Monday, Tuesday, Thursday and Friday from 10:00 am to 8:00 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Follansbee can be reached on (571) 272-3964. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Haresh Patel

September 13, 2006

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

1. (Currently amended) A system for streaming a software application to a client, the system comprising:

an application library having stored therein a prediction model and application files of the software application, the application files including executable code;

a streaming manager configured to send the application files to a client as a plurality of streamlets including executable code, each streamlet corresponding to a particular data block in a respective application file, and configured to send to the client, upon a first initiation of streaming the application, a file structure specification of the application files and a set of streamlets comprising at least those streamlets containing the portions of the application required to enable execution of the application to be initiated; and

a streaming prediction engine configured to identify at least one streamlet as most appropriate to send to the client at a particular time, based on an order in which code is predicted to be used when the software application is executed on the client, in accordance with the prediction model.

2. (Original) The system of claim 1, wherein each streamlet corresponds to a file data block having a size equal to a code page size used during file reads by an operating system expected to be present on a client system.

3. (Original) The system of claim 2, wherein the data block size is four kilobytes.

4. (Original) The system of claim 1, wherein the application files are stored in the application library as preprocessed streamlets, each streamlet corresponding to a data block in a particular application file at a particular offset and having a predefined length.

5. (Original) The system of claim 4, wherein the predefined length comprises a code page size used during file reads by an operating system expected to be present on a client system.

6. (Original) The system of claim 4, wherein each preprocessed streamlet is compressed.

7-8. (Canceled)

9. (Currently amended) The system of claim ~~8~~1, wherein the application library has a startup block comprising the file structure specification and set of streamlets stored therein.

10. (Original) The system of claim 1, wherein the streaming manager is further configured to install streaming environment support software on the client prior to initiating an application streaming processes.

11. (Original) The system of claim 1, further comprising a differential prediction model associated with the client, the prediction engine configured to make streamlet predictions for the client in accordance with the default prediction model and the respective differential prediction model.

12. (Original) The system of claim 11, wherein the streaming manager is configured to, upon receipt of application usage tracking information from the client, update at least one of the differential prediction model for the client and the prediction model.

13. (Original) The system of claim 1, further comprising an application status repository comprising a data map for each active client, the data map generally indicating the streamlets which are present at the respective client.

14. (Original) The system of claim 13, wherein the streaming manager is configured to update the data map for the client upon a successful transmission of a streamlet to the client.

15. (Original) The system of claim 14, wherein the streaming manager is configured to, upon receipt of a request for a particular streamlet from the client:

determine if the data map indicates that the client already has the requested streamlet;

if the data map indicates that the requested streamlet is on the client system, request an updated data map from the client and replace the data map with a returned updated map;

retrieve the requested streamlet from the application library; and

update the data map upon a successful transmission of the requested streamlet to the client.

16. (Original) The system of claim 15, wherein the streaming manager is further configured to, upon receipt of the streamlet request from the client, reposition the

prediction engine in the default prediction model in accordance with the requested streamlet.

17. (Original) The system of claim 13, wherein the streaming manager is configured to, upon receipt of an unsolicited data map from the client, replace the data map in the application status repository for the client with the data map received from the client.

18. (Original) The system of claim 17, wherein the streaming manager is further configured to, upon receipt of the unsolicited data map, compare the data map in the application status repository for the client with the data map received from the client and log mismatches.

19. (Currently amended) A method for streaming a software application, the method comprising the steps of:

providing at a server an application library having stored therein application files of the software application, the application files including executable code;

forwarding the application files to a client as a particular sequence of streamlets including executable code, each streamlet corresponding to a particular data block in a respective application file, including sending to the client, upon a first initiation of streaming the application, a file structure specification of the application files and a set of streamlets comprising at least those streamlets containing the portions of the application required to enable execution of the application to be initiated; and

determining the particular sequence of streamlets in accordance with a prediction model indicating which streamlets are most appropriate to send to the client at a

particular time, based on an order in which code is predicted to be used when the software application is executed on the client.

20. (Original) The method of claim 19, wherein each streamlet corresponds to a file data block having a size equal to a code page size used during file reads by an operating system expected to be present on a client system.

21. (Original) The method of claim 20, wherein the data block size is four kilobytes.

22. (Original) The method of claim 19, further comprising the step of dividing the application files into streamlets prior to initiation of a streaming session.

23. (Original) The method of claim 19, further comprising the step of storing the application files in the application library as preprocessed streamlets, each streamlet corresponding to a data block in a particular application file at a particular offset and having a predefined length.

24. (Original) The method of claim 23, wherein the predefined length comprises a code page size used during file reads by an operating system expected to be present on a client system.

25. (Original) The method of claim 23, further comprising the step of compressing each streamlet prior to storage in the application library.

26-27. (Canceled)

28. (Currently amended) The method of claim ~~27~~19, further comprising the step of storing in the application library a startup block comprising the file structure specification and set of streamlets stored therein.

29. (Original) The method of claim 19, further comprising the step of initiating a process to install streaming environment support software on the client prior to initiating an application streaming processes.

30. (Original) The method of claim 19, wherein the step of determining comprising determining the particular sequence of streamlets in accordance with the prediction model and a differential prediction model associated with the client.

31. (Original) The method of claim 30, further comprising the step of, upon receipt of application usage tracking information from the client, updating at least one of the differential prediction model for the client and the prediction model.

32. (Original) The method of claim 19, further comprising the steps of, upon receipt of a request for a particular streamlet from the client:

retrieving the requested streamlet from the application library; and

transmitting the streamlet to the client.

33. (Original) The method of claim 19, further comprising the steps of:

providing a data map for each active client generally indicating the streamlets which are present at the respective client; and

updating the data map associated with a particular client upon a successful transmission of a streamlet to the particular client.

34. (Original) The method of claim 33, further comprising the steps of, upon receipt of a request for a particular streamlet from the client:

determining if the data map associated with the client indicates that the already has the requested streamlet; and

in response to a positive determination, requesting an updated data map from the client and replacing the data map with a returned updated map.

35. (Original) The method of claim 34, further comprising the step of adjusting a position in the prediction model for the client in accordance with the requested streamlet.

36. (Original) The method of claim 33, further comprising the step of, upon receipt of an unsolicited data map from the client, replacing the data map in the application status repository for the client with the data map received from the client.

37. (Original) The method of claim 36, further comprising the steps of:

comparing the data map in the application status repository for the client with the unsolicited data map received from the client; and

logging mismatches identified during the comparing step.

38. (Currently amended) A computer program product stored on a computer readable medium, the product comprising a computer program for configuring a server with an

application library having application files including executable code stored therein to stream a software application to a client, the computer program comprising code to configure the server to:

forward the application files to the client as a particular sequence of streamlets including executable code, each streamlet corresponding to a particular data block in a respective application file, including sending to the client, upon a first initiation of streaming the software application, a file structure specification of the application files and a set of streamlets comprising at least those streamlets containing the portions of the software application required to enable execution of the software application to be initiated; and

determine the particular sequence of streamlets in accordance with a prediction model indicating which streamlets are most appropriate to send to the client at a particular time, based on an order in which code is predicted to be used when the software application is executed on the client.

39. (Original) The computer program product of claim 38, the computer program further comprising code to further configure the server to divide the application files into streamlets prior to initiation of a streaming session.

40. (Original) The computer program product of claim 39, the computer program further comprising code to configure the server to divide the application files into streamlets corresponding to a data block in a particular application file at a particular offset and having a predefined length.

41-42. (Canceled)

43. (Currently amended) The computer program product of claim 4238, the computer program further comprising code to store in the application library a startup block comprising the file structure specification and set of streamlets stored therein.

44. (Original) The computer program product of claim 38, the computer program further comprising code to install streaming environment support software on the client prior to initiating an application streaming processes.

45. (Original) The computer program product of claim 38, the computer program further comprising code to determine the particular sequence of streamlets in accordance with the prediction model and a differential prediction model associated with the client.

46. (Original) The computer program product of claim 45, the computer program further comprising code to, upon receipt at the server of application usage tracking information from the client, update at least one of the differential prediction model for the client and the prediction model.

47. (Original) The computer program product of claim 38, the computer program further comprising code to, upon receipt at the server of a request for a particular streamlet from the client:

retrieve the requested streamlet from the application library; and

transmit the streamlet to the client.

48. (Original) The computer program product of claim 38, the computer program further comprising code to:

provide a data map for each active client generally indicating the streamlets which are present at the respective client; and

update the data map associated with a particular client upon a successful transmission of a streamlet to the particular client.

49. (Original) The computer program product of claim 48, the computer program further comprising code to, upon receipt at the server of a request for a particular streamlet from the client:

determine if the data map associated with the client indicates that the already has the requested streamlet; and

in response to a positive determination, request an updated data map from the client and replacing the data map with a returned updated map.

50. (Original) The computer program product of claim 49, the computer program further comprising code to adjust a position in the prediction model for the client in accordance with the requested streamlet.

51. (Original) The computer program product of claim 48, the computer program further comprising code to, upon receipt at the server of an unsolicited data map from the client, replace the data map in the application status repository for the client with the data map received from the client.

52. (Original) The computer program product of claim 51, the computer program further comprising code to:

compare the data map in the application status repository for the client with the unsolicited data map received from the client; and

log mismatches identified during the comparing step.

53-68. (Canceled)